Unsupervised classification of multivariate geostatistical data: Two algorithms

Thomas Romary, Fabien Ors, Jacques Rivoirard, Jacques Deraisme

Unsupervised classification of multivariate geostatistical data: two algorithms

Thomas Romary*[1], Fabien Ors[1], Jacques Rivoirard[1], Jacques Deraisme[2]

* *corresponding author, thomas.romary@mines-paristech.fr*
[1] *MINES ParisTech, PSL Research University, Centre de Géosciences/Géostatistique, 35 rue Saint-Honoré, 77305 Fontainebleau, France*
[2] *Geovariances, 49 avenue Franklin Roosevelt, 77212 Avon, France*

Abstract

With the increasing development of remote sensing platforms and the evolution of sampling facilities in mining and oil industry, spatial datasets are becoming increasingly large, inform a growing number of variables and cover wider and wider areas. Therefore, it is often necessary to split the domain of study to account for radically different behaviors of the natural phenomenon over the domain and to simplify the subsequent modeling step. The definition of these areas can be seen as a problem of unsupervised classification, or clustering, where we try to divide the domain into homogeneous domains with respect to the values taken by the variables in hand. The application of classical clustering methods, designed for independent observations, does not ensure the spatial coherence of the resulting classes. Image segmentation methods, based e.g. on Markov random fields, are not adapted to irregularly sampled data. Other existing approaches, based on mixtures of Gaussian random functions estimated via the Expectation-Maximization algorithm, are limited to reasonable sample sizes and a small number of variables. In this work, we propose two algorithms based on adaptations of classical algorithms to multivariate geostatistical data. Both algorithms are model free and can handle large volumes of multivariate, irregularly spaced data. The first one proceeds by agglomerative hierarchical clustering. The spatial coherence is ensured by a proximity condition imposed for two clusters to merge. This proximity condition relies on a graph organizing the data in the coordinates space. The hierarchical algorithm can then be seen as a graph-partitioning algorithm. Following this interpretation, a spatial version of the spectral clustering algorithm is also proposed. The performances of both algorithms are assessed on toy examples and a mining dataset.

Keywords: Spatial data, clustering, multivariate data, geostatistics

1. Introduction

In mining assessment, a partitioning of the data is often to be conducted prior to evaluate the reserves. This is necessary to design the mineralization enveloppes corresponding to several lithofacies where the grades of the ore to be mined may have different spatial behavior, in terms of mean, variability and spatial structure. In remote sensing of environmental variables, a similar problem may be encountered when the characteristics of the variable of interest is governed by a hidden variable, e.g. the component of a mixture model, accounting for a particular local behaviour. A typical example in soil sciences consists in the the retrieval of soil classes over a region from the observation of continuous variables.

A natural solution to this problem is to cluster the data. Clustering a dataset consists in partitioning the observations into subsets (called clusters) so that observations in the same cluster are similar in some sense. Clustering is used in many fields, including machine learning, data mining, pattern recognition, image analysis, information retrieval and bioinformatics (Hastie et al., 2009). It is an unsupervised classification problem where the goal is to determine a structure among the data, with no response variable to lead the process.

While a wide range of methods exist for independent (Hastie et al., 2009) or gridded spatial observations (in the image processing litterature), not much attention has been paid to the case of irregularly spaced data. Indeed, in a geostatistical context, one expects to obtain a classification of the data that presents some spatial continuity. This is especially the case with mining data, where the geologist wishes to delineate homogeneous areas in a deposit to facilitate its evaluation and exploitation.

Clustering in a spatial framework has been mainly studied in the image analysis context where the data is organized on a grid. The model is usually a hidden Markov random field. In this model, label properties and pixel values need only to be conditioned on nearest neighbors instead of on all pixels of the map, see e.g. Guyon (1995) for a review and Celeux et al. (2003) for more recent developments. In Ambroise et al. (1995), the authors proposed to use this approach directly to irregularly sampled data using a neighborhood defined by the Delaunay graph of the data. As the length of the edges of the graph are not accounted for in the approach, this neighborhood structure does not reflect a structure in the data, rather a structure in

the sampling scheme. This disqualifies this approach especially for mining data, where the samples are located along drillholes: two neighbors on a same drillhole are distant a few centimeters while two neighbors from two different drillholes may be distant several decimeters.

Oliver and Webster (Oliver and Webster, 1989) were the first to propose a method for the clustering of multivariate non-lattice data. They proposed to modify the dissimilarity matrix of the data, used e.g. in a hierarchical algorithm, by multiplying it by a variogram matrix. This terms to smooth the dissimilarity matrix for close pairs of points. However, this will not enforce the connexity of the resulting clusters, it will rather blur the borders between geologically different areas, making them difficult to differentiate, as our practice showed.

In Allard and Guillot (2000), the authors proposed a clustering method based on a mixture of random functions models where an approximation of the expectation-maximization (EM, see Dempster et al., 1977) algorithm is used to estimate the parameters and the labels. It has been later extended to multivariate data in Guillot et al. (2006). However this method relies on strong assumptions that are not likely to be encountered in practice: the data are assumed to be Gaussian and data belonging to different clusters are assumed independent. Moreover, the estimation algorithm requires the computation of the maximum likelihood estimator of the random function model at each iteration of the EM algorithm, which involves the inversion of the covariance matrix and is not computationally tractable for large, multivariate datasets. Indeed, a single iteration requires several inversions of a $(n \times p) \times (n \times p)$ matrix, where n is the number of data and p is the number of variables. Using composite likelihood techniques (Varin et al., 2011) could be useful to alleviate the computational burden but it will add a degree of approximation while still not allowing to deal with categorical data.

The approaches developped in this paper are model free and do not involve complex computations. Therefore, they are able to process large, multivariate datasets. The first one, already outlined in Romary et al. (2012), is based on an agglomerative hierarchical algorithm with complete linkage (see e.g. Hastie et al., 2009), where the connexity of the resulting clusters is enforced through the use of a graph structuring the data. It only involves the computation of distances along the edges of the graph which has a sparse structure. Its sequential nature makes it practical for reasonable volumes of data. An alternative for large datasets consists however in running first the algorithm on a subsample, then training a supervised classifier and finally applying it to the rest of the data. The second proposed algorithm provides a non-hierarchical alternative to partition the same graph. It is an adap-

tation of the spectral clustering algorithm (Ng et al., 2002; von Luxburg, 2007) to geostatistical data. The computations involve only sparse matrices, therefore this second algorithm is adapted to large volumes of data.
The paper is organized as follows: in section 2, we describe both algorithms as well as a method to classify newly available data based on the results of a preceding clustering. In section 3, we show the performance of each algorithm on a synthetic dataset as well as on a mining dataset.

2. Algorithms

Both algorithms proposed rely on the same basic idea. The latter consists in structuring the available data in a graph in the geographical space made of a unique connex component. This graph is then partitioned into clusters either hierarchically or directly by decomposition. The structure thus imposed ensures the spatial coherency of the resulting clusters.
We consider a sample of georeferenced data $(x_1, \ldots, x_n) \in \mathbb{R}^{n \times p}$, where p is the number of variables, coordinates included. We also consider that the data have been standardized preliminary to the application of the clustering algorithms. It may also be useful to gaussianize the variables, e.g. by anamorphosis (Chilès and Delfiner, 2012), for skewed data. This preliminary processing allows to make the variables comparable. We describe in this section the different ingredients required to implement both algorithms as well as their core.

2.1. Structuring the data

Being either regular or not, the spatial sampling of a geostatistical dataset defines a geometric set, namely a set of points in the geographical space. From this set, a neighborhood system can be built. This can be represented by an undirected graph where each vertex represents an observation and each edge shows the relation of neighborhood shared by close points (Geman and Geman, 1984). We call this graph the sampling graph. Several methods can be applied to build it such as Delaunay triangulation, Gabriel graph or a graph based on the neighborhood structure defined by moving neighborhood algorithms used in kriging, for instance based on octants (see e.g. Chilès and Delfiner, 2012). Particular shapes can also be obtained by using non-isotropic distances or coordinates transformation. The graph should be parsimonious whilst containing enough edges to support a variety of configurations for the clusters. In our experience, the Delaunay graph and a graph based on a neighborhood selection algorithm give good results. Once the graph G has been built, two observations x_i and $x_j, i \neq j$, are said

to be connected if their exists an edge in G linking x_i and x_j. This is denoted by $x_i \leftrightarrow x_j$. G can also be represented by its adjacency matrix with general term $(G_{ij})_{i,j\in\{1,\dots,n\}}$:

$$G_{ij} = \begin{cases} 1 & \text{if } x_i \leftrightarrow x_j, \\ 0 & \text{otherwise.} \end{cases}$$

Note that an individual is not considered to be connected with itself.

2.2. Choice of a distance

The second basic ingredient of both algorithms is a distance or metric measuring the dissimilarity between two observations. The aim of clustering algorithms is to group similar observations, hence the need to define *similar*. We define the distance d between two observations x_i and x_j by:

$$d(x_i, x_j) = \sum_{k=1}^{p} \sum_{l=1}^{p} \omega_{k,l} d^{(k,l)}(x_i^{(k)}, x_j^{(l)}), \tag{1}$$

where $(\omega_{k,l})_{(k,l)=(1,\dots,p)^2}$ are the entries of a positive definite matrix Ω and $(d^{(k,l)})_{k=1,\dots,p}$ is a set of coupled distances, each one chosen according to the corresponding couple of variables. d is therefore a weighted sum of distances. The weights are to be chosen by the user, depending on the relative importance the variables should have and their possible correlation. As noted above, the variables have been preliminary standardized so as to avoid any scale effect between the variables. In practice, Ω is generally chosen to be diagonal and only individual distances are thus involved. The use of the squared Mahalanobis distance, where Ω is the inverse of the empirical covariance matrix could be considered so as to account for possible correlations between variables, but has not proven useful in our experiments. The individual distances are chosen according to the associated variable: if the latter is quantitative, the squared euclidean distance is advocated from its strong relation with the variogram as a measure of the local continuity; if it is a categorical variable, an ad-hoc distance is used. Such a distance may take the value 0 when both observations have an equal value for this variable and 1 otherwise. Other options are also available, see e.g. Hastie et al. (2009) for a comprehensive view.

It is worth noting that the coordinates are also included in (1). Indeed, although the spatial location of the data is already accounted for by the graph structure, this allows to account for the length of the edges. By doing this, we promote short connections.

Concerning the setting up of the weights, we generally recommend to put 5% to 30% of the total on the coordinates and to set the other variables to 1 at a first guess, then to progressively tune the weights of the variables according to the outcome of the algorithm.

2.3. Geostatistical Hierarchical Clustering

The distance defined above is only valid between pairs of observations. Agglomerative hierarchical clustering algorithms require a linkage criterion which specifies the dissimilarity of sets as a function of the pairwise distances of observations in the sets. Lance and Williams formula (Lance and Williams, 1966) enables the use of a unique recurrence formula to update the distances when merging two clusters for a large family of criteria, including the maximum, minimum or average distance, respectively named complete, single or average linkage criteria or Ward's criterion which computes the intra-cluster variance, see e.g. Milligan (1979).

In our context, the spatial continuity needs to be taken into account during the linkage process. In the proposed algorithm, two clusters can merge if and only if they are connected in the graph structure G. When two clusters merge, the resulting cluster inherits all connections of its components. This point is the only departure from the original hierarchical clustering algorithm.

The geostatistical hierarchical clustering algorithm (GHC) is described in pseudo code in algorithm 1 under the complete linkage criterion.

Algorithm 1 Geostatistical Hierarchical Clustering algorithm (GHC)

1: Compute the distance matrix $D \in \mathbb{R}^{n \times n}$, such that $D_{ij} = d(x_i, x_j),\ j < i$, if $i \leftrightarrow j$, 0 otherwise
2: **repeat**
3: Find k and l, $k < l$, such that $D_{lk} = \min_{\{i,j,i \leftrightarrow j\}} D_{ij}$
4: Merge k and l in $\{kl\}$, and update D such that

$$D_{ki} = \max(D_{ki}, D_{li}) \text{ if } i \leftrightarrow \{kl\} \text{ and } i < k$$
$$D_{ik} = \max(D_{ik}, D_{li}) \text{ if } i \leftrightarrow \{kl\} \text{ and } k < i < l$$
$$D_{ik} = \max(D_{ik}, D_{il}) \text{ if } i \leftrightarrow \{kl\} \text{ and } i > l$$

discard line and column l from D
5: **until** D is a scalar

In algorithm 1, the value D_{lk} can be interpreted as the inner distance or dissimilarity of the cluster obtained when merging clusters k and l. The

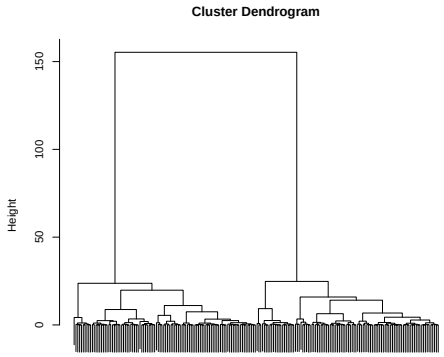


Figure 1: Example of a dendrogram

notation $i \leftrightarrow \{kl\}$ means i is connected with the cluster $\{kl\}$, that is $i \leftrightarrow k$ or $i \leftrightarrow l$.

Since two clusters are merged when they realize the minimum distance among the connected pairs of clusters, they may not realize the minimum distance in absolute, depending on the chosen linkage criterion. In particular, more dissimilar points may merge into clusters before having merged points which are actually more similar but not directly connected. That is why we advocate the use of the complete linkage criterion which is, to our knowledge, the only way to preserve the ultrametric property in our algorithm. The ultrametric property means a monotonic increase of the dissimilarity value of the clusters, see Milligan (1979) for further details. In particular, the ultrametric property allows to build a dendrogram.

The dendrogram is a very practical tool to select the final number of clusters, see an example in figure 1. It represents the evolution of the intra-cluster dissimilarity along the agglomeration process. A long branch means that the merge between two clusters leads to a much less homogeneous one. Therefore the tree should be pruned at the level where the branches are long. The number of pruned branches gives the number of clusters to consider, 2 in the example of figure 1.

The computational efficiency of this algorithm relies on the graph structure employed and especially on the number of connections. Indeed, only the distances between connected points are required at the beginning of the

algorithm, which makes the matrix D sparse and allows fast computations. Then, the computation of the distances between connected points required at step 4 can be performed on the fly.

2.4. Geostatistical Spectral Clustering

The Geostatistical Spectral Clustering (GSC) is an adaptation of the spectral clustering algorithm where the graph used is the sampling graph defined above instead of a graph based on the similarity. Contrarily to GHC, it requires a preselection of the number K of desired clusters and does not rely on an iterative procedure. This is not a major drawback however. Once computed the quantities required for a given maximum number of classes, it is straightforward to compute the outcome for a smaller number of classes. The different steps of the algorithm are described in algorithm 2.

Algorithm 2 Geostatistical Spectral Clustering algorithm (GSC)

1: Compute the similarity or weighted adjacency matrix W:

$$W_{ij} = \begin{cases} \exp\left(-\frac{d(x_i,x_j)}{\sigma^2}\right) & \text{if } i \leftrightarrow j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

2: Compute the degree matrix D:

$$D_{ii} = \sum_{j=1}^{n} W_{ij}$$

3: Compute the graph Laplacian matrix

$$L = D^{-1/2} W D^{-1/2}$$

4: Compute the K largest eigenvalues of L and form the matrix $V \in \mathbb{R}^{n \times K}$ whose columns are the associated K first eigenvectors of L
5: Apply the K-means algorithm to the lines of V
6: Assign observation x_i to the same class the line i of V has been assigned

This algorithm consists in representing the data into an infinite dimensional space (the reproducing kernel Hilbert space associated to the kernel used in (2), here the Gaussian (or radial basis function kernel) where they are easily clustered through K-means.

The parameter σ^2 is chosen as the empirical variance of the variable, following von Luxburg (2007). Note that a local adaptive approach could be considered for the setting of σ^2, as proposed in Zelnik-Manor and Perona (2004). However, this refinement has not proven useful in our practice. Also, the lines of V can be optionally normalized prior to step 5, as proposed in Ng et al. (2002). The differences when using the normalization or not did not appear sensible in our experimentations.

The number of clusters to consider can be chosen by studying the eigenvalues of L. A small eigenvalue signifies that the associated eigenvector is not relevant to discriminates the data. In practice, we advocate to compute a given maximum number of eigenvalues (10 to 20), which corresponds to the maximum number of clusters we want, and then to plot them. A large difference between two eigenvalues means that the smaller one is not so relevant.

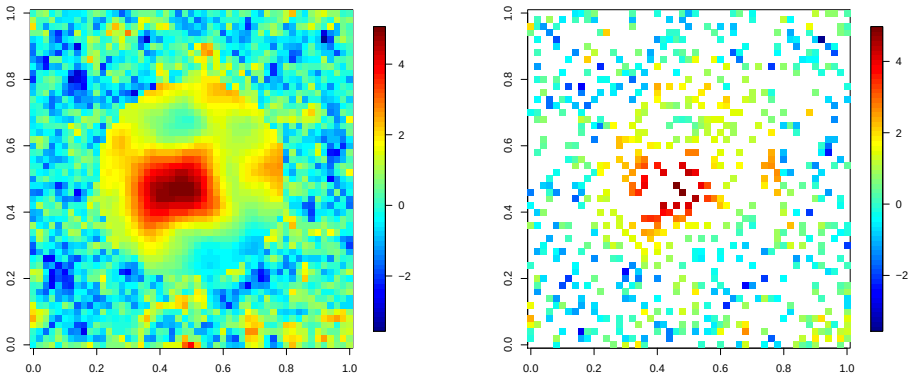
As the graph structure is sparse, all the computations required in algorithm 2 can be carried out using sparse linear matrix algebra, which makes GSC computationally efficient and adapted to large multivariate datasets.

2.5. Classifying new data

Sometimes, the sampling of the variables of interest on a domain can be performed in several steps. For instance, new drilholes can be added to an initial sampling campaign. In the case where a clustering has already been performed, we may want to classify the new data into the classes resulting from that previous run. An other occurrence when we want to classify data upon the results of a previous clustering is when dealing with very large datasets with the GHC. In that case, we propose to run first the algorithm on a subsample, then train a supervised classifier and finally apply the latter to the remaining data.

It is particularly difficult to incorporate new data into the clustering results with simple rules. Indeed, when new data are added, the sampling graph gets modified and the outcome of GHC and GSC may change dramatically. Therefore, the idea developed here is to learn a classification rule based on the initial clustering results. This can be achieved for instance through support vector machines (SVM, see Hastie et al. (2009)). In the case of two classes, the basic principle is to find $f(x) = \alpha_0 + \sum_{i=1}^{N} \alpha_i \Phi(x, x_i)$, where the $(\alpha_i)_{i=0,\dots,N}$ are scalars and Φ a given kernel function, that minimizes

$$\sum_{i=1}^{N} (1 - y_i f(x_i))_+ + \lambda \alpha^t \Phi \alpha, \tag{3}$$



a. *b.*

Figure 2: One realization of the random function *a.* and sampling performed *b.*

as a function of $(\alpha_i)_{i=0,\dots,N}$ and where the underscript + means the maximum between 0 and the quantity between parenthesis, and λ is a penalty parameter. For multi-class classification, several options are available among which we retain the standard "one versus all" implemented in LIBSVM (Chang and Lin, 2011). The penalty parameter λ is set through cross-validation. Applying the rule to a new observation allows to assign it to an existing class.

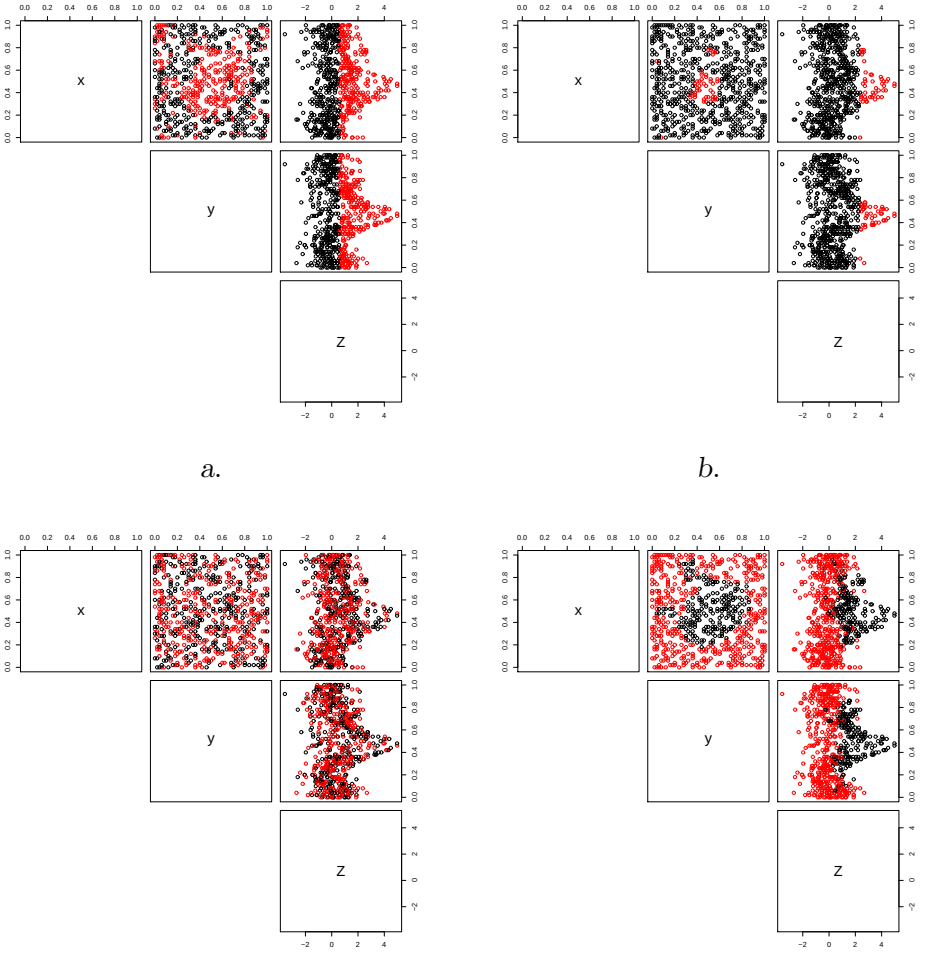
3. Results

3.1. Toy dataset

Here, we describe a 2D example on which we have evaluated the performances of several methods including GHC and GSC. We consider a random function on the unit square which is made of a Gaussian random function with mean 2 and a cubic covariance with range 0.3 and sill 1 on the disk of radius 0.3 and center (0.5,0.5) and a Gaussian random function with mean 0 and an exponential covariance with range 0.1 and sill 1 elsewhere. This model is made to mimick a mineralization area in a mining deposit, where high grades are more likely to be found within the disk. A realization is shown in figure 2 *a.* while figure 2 *b.* corresponds to the sampling performed by picking 650 points out of the 2601 points of the complete realization.

We can clearly see a smooth surface with high values in the central disk in figure 2 *a.* and this is the area we would like to retrieve from the 650 observations plotted in figure 2 *b.*. We test the performances of five different methods for this task: K-means, complete linkage hierarchical clustering (HC), Oliver and Webster's method (O&W), GHC and GSC.
For every method, the three variables are scaled such that the coordinates are given a weight of 10% in the computation of the distance. This preliminary treatment makes the different methods comparable. In HC, O&W, GHC and GSC, we use the squared euclidean distance. K-means does not need any parameterization. For O&W, several variogram models and sets of parameters have been considered, without much success. The results presented here are obtained with an exponential variogram with range 0.5. The Delaunay graph has been used for both GHC and GSC. Concerning GSC, normalizing the rows of V (see algorithm 2) gave similar results as without normalization. Consequently, only the results without normalization are presented.

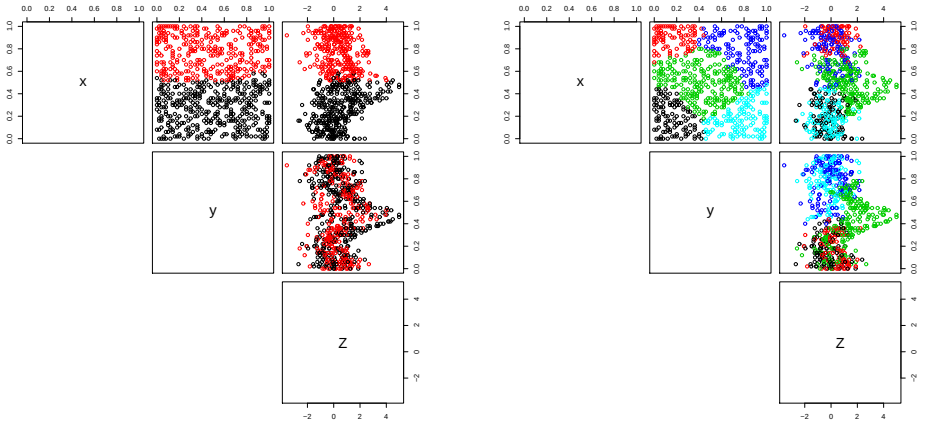
Figures 3 and 4 show the results obtained by each five methods on the realization depicted in figure 2. Each subpicture represents the dataset on scatterplots with respect to the coordinates (x and y) and the sampled value (Z). K-means (*a.*) identifies well the central area. The result lacks of connexity however. In particular, large values outside of the disk are classified as belonging to the disk and low values within the disk are missclassified as well. It can be seen that the method only discriminates between low and high values of Z: the limiting value between the two clusters can be read as more or less 0.5. HC (*b.*) also discriminates between low and high value but the limiting value is higher, around 2. To sum up, those two classical methods in an independent observations context fail to produce spatially connected clusters. O & W's approach has been tested with various variograms and variogram parameter values but it never showed any structured result (*c.*). Our interpretation is that multiplying the dissimilarity matrix by a variogram may erase some dissimilarities, inducing a loss in the structure of the data. The GHC algorithm succeeds in providing a clustering with spatial connexity (*d.*) though non perfect. A part of the area surrounding the disk is misclassified however. If we turn back to the complete realization in figure 2 *a*, we can see that the misclassified area corresponds to high values of the realization around the border of the disk that are very close to the values taken inside the disk and are thus difficult to classify correctly. Finally, the GSC algorithm performed a geometrical classification by making a cut along the axis of the first coordinate, see figure 4. However, when looking at the result obtained when asking for five classes, it provided



a. *b.*

c. *d.*

Figure 3: Results of K-means *a.*, hierarchical clustering *b.*, Oliver and Webster's method *c.* and geostatistical hierarchical clustering *d.*



a. *b.*

Figure 4: Results of GSC for two *a.* and five classes *b.*

a class delineating the disk fairly well. It seems that this algorithm tends to generate more compact subsets of the sampling graph.

Each of the five algorithms are applied to 100 realizations of the same random function model, each with a different uniform random sampling. Then we compute the mean, median and 90% percentile of the rate of correctly classified points. Results are summarized in table 1.

GHC exhibits the best performances overall with 85% correctly classified points in average while K-means providing similar results in average, GSC performing the worst with HC and O & W in between. If we look at the median however, GHC has the greatest one with a larger margin. The 90% percentile indicates that in the 10% most favorables cases, GHC misclassified only 0.02% of the points, while all the other algorithms perform worse. It can also be seen that the 90% percentile are similar for the K-means and the HC. This means that the HC, and GHC (its worse result in this task was a misclassification of almost 50%, seemingly due to a high sensitivity to large values), can sometimes perform really bad, whereas the K-means algorithm gives more stable results, being less sensitive to extreme values. Indeed, in the presence of very large or very low value, it occurs that the algorithm comes out with a class made of a single point while the other contains all the other observations. In the favorable cases however, HC algorithm works as well as the K-means, while GHC outperforms clearly all other algorithms. Concerning GSC, the results obtained are extremely poor

	K-means	HC	O & W	GHC	GSC
Mean	0.86	0.70	0.65	0.85	0.52
Median	0.86	0.64	0.67	0.90	0.52
90% percentile	0.90	0.91	0.72	0.98	0.54

Table 1: Rates of correctly classified points for the 5 algorithms

but do not account for the interesting results obtained when considering more classes.
It is worth noting that the drawbacks exhibited by GHC and GSC are far from being prohibitive in practice. Indeed, when applying clustering algorithms to real data the user generally observes the outcome for several numbers of classes. This can be performed easily with both algorithms with a negligible computational cost.

3.2. Mining data example

In this section, we present an application of both geostatistical clustering algorithms to an ore deposit. We describe the different steps and exhibit some results.
The first step is to select the data that will be used for the clustering. The following variables are chosen:

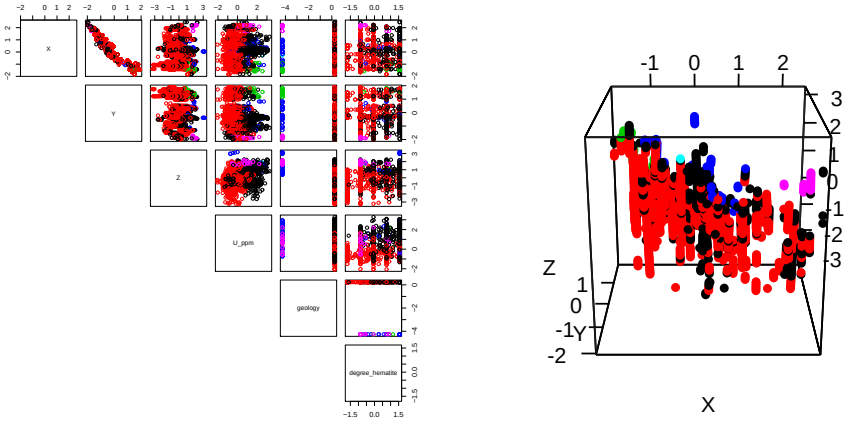
- coordinates, X, Y and Z,
- ore grades,
- a geological factor describing the basement vs. a sandy part on top of it,
- the hematization degree.

This choice is made upon an exploratory analysis of the data and discussions with geologists. Some transformations of the data are preliminary performed:

- coordinates are standardized,
- ore grades are log-transformed and standardized,
- the degree of hematization is transformed into a continuous variable, then standardized.

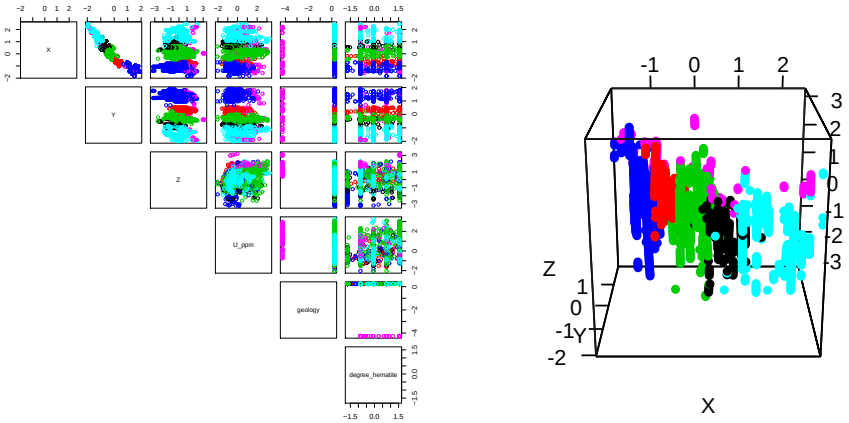
The next step consists of building the sampling graph connections between geographically close samples. The graph is here built from the neighbouring structure induced by the moving neighbourhood kriging algorithm of Isatis® 2013 (Geovariances, 2013). At each point, the space is split into 16 hexadecants: 8 above and 8 below the horizon. One neighbor per hexadecant is authorized at most for each point with no more than 2 from the same drillhole. The search ellipse is of infinite size so as to connect even possibly distant points. The angles of the search ellipse are chosen so that to take into account the horizontal shape of the mineralization of the deposit.
Then the dissimilarity matrix is built. All variables listed above are used. A particular distance for the geological factor is considered: it is chosen to be 1 when the samples have different factor values and 0 otherwise. This distance is scaled to maintain the coherency with the other individual distances. Weights are set step by step, as advocated in section 2.2: we begin by giving an equal weight to all variables with a 30% contribution to the coordinates. Finally, the contribution of the coordinates is lowered to 10% while the other variables are assigned equal weights. The same set of weights is used for both algorithms. Practice shows indeed that setting low weights to the coordinates leads to better results, as the spatial aspect is already somehow taken into account by the sampling graph. However, the coordinates needs to be included in the distance so as to account for different length of the edges in the graph. This is especially important for drillholes data where two neighbors along a drillhole are generally much closer than two neighbors belonging to two different drillholes.
Finally, we can run both GHC and GSC algorithms described in section 2. We choose to represent 6 clusters as the intra cluster dissimilarity at that step of the GHC shows a great increase. The results are depicted in figure 5 for GHC and 6 for GSC.

GHC separates the basement into two classes, the black one being richer than the red one. Note that the black cluster is mainly present in the middle of the deposit. The sandy part on top of the basement is splitted into 3 separate classes plus one single observation (in cyan), see figure 5. The discrimination between the three sandy classes seems to rely on geographical considerations.
As for GSC, it splits the basement into 5 classes and puts every observation on top of it into one single class. Some similarities can be observed between the clustering results obtained with the two algorithms however. In particular, both make a clear distinction between the basement and the sand on top of it, emphasizing the variable 'geology'. They also both exhibit the desired connexity properties. Both also reveal a high grades area in the center of



a. *b.*

Figure 5: Resulting clusters for the GHC algorithm from the variables point of view *a.* and in 3D *b.*



a. *b.*

Figure 6: Resulting clusters for the GSC algorithm from the variables point of view *a.* and in 3D *b.*

the deposit (the black cluster in both figures), whose retrieval was the goal of the experimentation. As already noticed in the previous paragraph, GSC tends to produce more compact clusters than GHC who can follow awkward routes along the graph.

4. Conclusion

In this paper, we presented two clustering procedures adapted to irregularly sampled spatial data. Both algorithms allow to process large multivariate datasets. They rely on a partition of a graph structuring the data in the geopraphical space, thus ensuring the spatial coherency of the resulting clusters. Two applications have been provided, the first one on a toy example and the second on mining data.

The results shown on the toy example validate both algorithms as they are able to produce compact, connected clusters. The results obtained for the mining application are also satisfactory as they highlight a homogeneous area with high grades. Thanks to the sequential nature of GHC, it generates a whole ensemble of coherent clusterings that can be useful to the user: he can visualize the results at different hierarchical levels which helps the interpretation and the choice of the final number of clusters for the end user. Note that GSC does not enjoy this property as the results may change dramatically from one desired number of clusters to another. The main drawback of GHC is its limitation to datasets of reasonable size. It becomes slow when the number of observations goes beyond 10000. In the case of large datasets, a two step approach based on subsampling and supervised classification is proposed.

Finally, setting the distance used to compute the graph and the weights associated to each variable allows the practitioner to get different clusterings, according to its knowledge of the geology and the variables he wants to be emphasized in the results. The main difficulty in handling these algorithms is their sensitivity to the different parameters used. Moreover the results are difficult to validate except from the computation of indices of compactness of the clusters or of heterogeneity between them. They are mostly to be validated by the eye of the practitioner whose knowledge of the data should guide in the step by step parameterization of the approach.

Acknowledgement

This work has been partly funded by the Geological and Geostatistical Domaining (G^2DC) consortium, conducted by Geovariances. The authors

are grateful to all members of the consortium for helpful discussions, namely Anglogold Ashanti, Areva, BHP Billiton, Eramet and Vale.

References

Allard, D., Guillot, G., 2000. Clustering geostatistical data. In: Proceedings of the sixth geostatistical conference.

Ambroise, C., Dang, M., Govaert, G., 1995. Clustering of spatial data by the EM algorithm. In: et al., A. S. (Ed.), geoENV I - Geostatistics for Environmental Applications. Kluwer Academic Publishers, pp. 493–504.

Celeux, G., Forbes, F., Peyrard, N., 2003. Em procedures using mean field-like approximations for markov model-based image segmentation. Pattern recognition 36 (1), 131–144.

Chang, C.-C., Lin, C.-J., 2011. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chilès, J. P., Delfiner, P., 2012. Geostatistics, Modeling Spatial Uncertainty, 2nd Edition. John Wiley & Sons, New-York.

Dempster, A. P., Laird, N. M., Rubin, D. B., 1977. Maximum likelihood from incomplete data via EM algorithm (with discussion). Journal of the Royal Statistical Society, Ser. B 39, 1–38.

Geman, S., Geman, D., 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. Pattern Analysis and Machine Intelligence, IEEE Transactions on (6), 721–741.

Geovariances, 2013. Isatis technical references. Version 13.

Guillot, G., Kan-King-Yu, D., Michelin, J., Huet, P., 2006. Inference of a hidden spatial tessellation from multivariate data: application to the delineation of homogeneous regions in an agricultural field. Journal of the Royal Statistical Society: Series C (Applied Statistics) 55 (3), 407–430.

Guyon, X., 1995. Random fields on a network. Springer.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning, 2nd Edition. Springer.

Lance, G., Williams, W., 1966. A generalized sorting strategy for computer classifications. Nature.

Milligan, G. W., 1979. Ultrametric hierarchical clustering algorithms. Psychometrika 44 (3), 343–346.

Ng, A., Jordan, M., Weiss, Y., 2002. On spectral clustering: analysis and an algorithm. In: Dietterich, T., Becker, S., Ghahramani, Z. (Eds.), Advances in Neural Information Processing Systems. Vol. 14. MIT Press, p. 849 – 856.

Oliver, M., Webster, R., 1989. A geostatistical basis for spatial weighting in multivariate classification. Mathematical Geology 21, 15–35, 10.1007/BF00897238.
URL http://dx.doi.org/10.1007/BF00897238

Romary, T., Rivoirard, J., Deraisme, J., Quinones, C., Freulon, X., 2012. Domaining by clustering multivariate geostatistical data. In: Geostatistics Oslo 2012. Springer, pp. 455–466.

Varin, C., Reid, N. M., Firth, D., 2011. An overview of composite likelihood methods. Statistica Sinica 21 (1), 5–42.

von Luxburg, U., 2007. A tutorial on spectral clustering. Statistics and Computing 17 (4).

Zelnik-Manor, L., Perona, P., 2004. Self-tuning spectral clustering. In: Advances in Neural Information Processing Systems 17. MIT Press, pp. 1601–1608.